DATA MINING OF MACHINE DESIGN ELEMENTS USING AI TECHNIQUE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology
In
Mechanical Engineering

By
SANDIP DATTA
&
SABIR KUMAR SAMAD

Department of Mechanical Engineering
National Institute of Technology
Rourkela
2007

DATA MINING OF MACHINE DESIGN ELEMENTS USING AI TECHNIQUES

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology
In
Mechanical Engineering

By
SANDIP DATTA
&
SABIR KUMAR SAMAD

Under the Guidance of
DR. D.R.K PADHI

Department of Mechanical Engineering
National Institute of Technology
Rourkela
2007

National Institute of Technology
Rourkela

CERTIFICATE

This is to certify that the thesis entitled, “Data Mining of Machine Design Elements Using AI Techniques” submitted by Sri Sandip Datta and Sri Sabir Kumar Samad in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Mechanical Engineering at the National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Date

Dr. D.R.K Padhi
Dept. of Mechanical Engineering
National Institute of Technology
Rourkela - 769008

National Institute of Technology
Rourkela

ACKNOWLEDGEMENT

I would like to articulate my deep gratitude to my project guide Dr. D.R.K Padhi who has always been my motivation for carrying out the project.

It is my pleasure to refer Microsoft word 2003 of which the compilation of this report would have been impossible.

An assemblage of this nature could never have been attempted without reference to and inspiration from the works of others whose details are mentioned in reference section. I acknowledge my indebtedness to all of them.

Last but not the least to all of my friends who were patiently extended all sorts of help for accomplishing this undertaking.

Date

Sabir Kumar Samad

Sandip Datta

Dept. of Mechanical Engineering

National Institute of Technology

Rourkela - 769008

CONTENTS

Sl. No.	Topic	Page
1	Certificate	i
2	Acknowledgement	ii
3	Contents	iii
4	Abstract	iv
5	Chapter 1 : Introduction Use of Data Mining Supporting Software	1 3 5
6	Chapter 2 : Basics of SQL Steps to Create a Database Query with the Database Link between Two Databases	8 9 11 14
7	Chapter 3 : Global.asa Nimstyle.css Nitrdesign.asp Nitrelements.asp Mainpage.html	18 19 20 22 24
8	Chapter 5 : Results and Output Scope for further Enhancement Conclusion	26 29 30
9	Reference	31

National Institute of Technology

Rourkela

ABSTRACT

Data Mining is the process of extracting knowledge, hidden from large volume of raw data. AI is about simulating human intelligence. The project aims at proving that Data mining methods can be realized in Mechanical Engineering Industries. To prove this we have created a Database containing design elements and have made the process to retrieved design elements as per the requirements. The same when applied to any industry, it can be proved beneficial and also cut down a lot of time wasted due to human in efficiency.

This includes several steps:

1. Design a Database which stores data,
2. using asp.net code to retrieve the data,

Data mining techniques viable tools for determining interesting patterns, clustering the parameter space, detecting anomalies in the simulation results, and for designing improved physical models. In mechanical industries, this technology can be used for a variety of jobs like forecasting, file management, providing information regarding the availability of material in production processes and also failure analysis in maintainance industries.

The project mainly deals with building a huge database, collecting data from data-book, using SQL server 2000. retrieval is done using asp.net programming on WINDOWS 2003 Enterprise Edition platform.

Chapter 1

INTRODUCTION

Introduction

Use of Data Mining

Supporting Software

INTRODUCTION

DATA MINING:

Data Mining is the process of extracting knowledge hidden from large volumes of raw data.

The importance of collecting data that reflect your business or scientific activities to achieve competitive advantage is widely recognized now. Powerful systems for collecting data and managing it in large databases are in place in all large and mid-range Companies. However, the bottleneck of turning this data into your success is the difficulty of extracting knowledge about the system you study from the collected data.

Data mining is an Al powered tool that can discover useful information within a database that can then be used to improve actions. To appreciate why businesses are so excited about data mining, you need only imagine that a major department store chain is looking for ways to boost sales. They have a large database containing information about customers and the nature of their purchases (with particulars such as identity of items, price, date, and time of sale). Suppose a data mining utility unearthed a pattern in the data which indicated that customers who shopped on Saturday afternoons and who made their initial purchase of the day in the shoe department tended to make, on average, 4 additional purchases from other departments and that the average member of this group spent more per visit than the typical shopper.

ARTIFICIAL INTELLIGENCE:

It is the science and engineering of making intelligent machines, especially intelligent computer programs. It is related to the similar task of using computers to understand human intelligence, but Al does not have to confine itself to methods that are biologically observable.

Intelligence involves mechanisms, and Al research has discovered how to make computers carry out some of them and not others. If doing a task requires only mechanisms that are well understood today, computer programs can give very impressive performances on these tasks, Such programs should be considered “somewhat intelligent”.

Al is about simulating human intelligence. On the one hand, we can learn something about how to make machines solve problems by observing other people or just by observing our own methods. On the other hand, most work in Al involves studying the problems the world presents to intelligence rather than studying people or animals, Al researchers are free to use methods that are not observed in people or that involve much more computing than people can do.

ABOUT THE PROJECT

This project is a very self illustrating application of data mining. The project is about using Artificial Intelligence tools to retrieve data from a data mine we have created. In this case we have made a data mine of around 2 lakh words, along with its meanings. Our aim is to use asp.net technology to access these words according to any random user’s choice. The same when applied to any industry can prove beneficial and also cut down a lot of time wasted due to human inefficiency. In mechanical industry this technology can be used for a variety of jobs like forecasting, file management, providing information regarding the availability of material in production processes and also failure analysis in maintenance industry.

Why use data mining?

Data might be one of the most valuable assets of any corporation - but only if you know how to reveal valuable knowledge hidden in raw data. Data mining allows you to extract diamonds of knowledge from your historical data and predict outcomes of future situations. It will help you optimize your business decisions, increase the value of each customer and communication, and improve satisfaction of customer with your services.

In all these cases data mining can help you reveal knowledge hidden in data and turn this knowledge into a crucial competitive advantage. Today increasingly more companies acknowledge the value of this new opportunity and turn to Megaputer for leading edge data mining tools and solutions that help optimizing their operations and increase your bottom line.

Reasons for the growing popularity of Data Mining

Growing Data Volume:

The main reason for necessity of automated computer systems for intelligent data analysis is the enormous volume of existing and newly appearing data that require processing. The amount of data accumulated each day by various business, production, scientific, and governmental organizations around the world is daunting. It becomes impossible for human analysts to cope with such overwhelming amounts of data.

Limitations of Human Analysis:

Two other problems that surface when human analysts process data are the inadequacy of the human brain when searching for complex multifactor dependencies in data, and the lack of objectiveness in such an analysis, A human expert is always a hostage of the previous experience of investigating other systems. Sometimes this helps, sometimes this hurts, but it is almost impossible to get rid of this fact.

Low Cost of Machine Learning:

One additional benefit of using automated data mining systems is that this process has a much lower cost than hiring an army of highly trained (and paid) professional statisticians. While data mining does not eliminate human participation in solving the task completely, it significantly simplifies the job and allows an analyst who is not a professional in statistics and programming to manage the process of extracting knowledge from data.

Tasks solved by Data Mining

A task of learning for the most influential independent variables for a selected target variable

Explicit modeling:-

A task, of finding explicit formulae, describing dependencies between various variables.

Clustering:-

A task of identifying groups of records that are similar between themselves but different from the rest of the data. Often, the variables providing the best clustering should be identified as well.

Market Basket Analysis:-

Processing transactional data in order to find those groups of products that is sold together well. One also searches for directed association rules identifying the best product to be offered with a current selection of purchased products.

Deviation Detection:-

A task, of determining the most significant changes in some key measures of data from previous or expected values.

SOFTWARE SUPPORT

The project has to be done in **"WINDOWS 2003 Enterprise Edition"** platform in order to invoke **asp.net** facilities with the help of Internet Information Services. A data base of words must be created, so we use a renowned database development tool, **SQL Server 2000.**

Why SQL server as database:

Until now, you had two choices: ignore the data you couldn't find or hire a statistician to apply algorithms to your data. That's all changed, due to the marriage of research and product groups at Microsoft. The Data Management, Mining and Exploration (DMMX) group integrated a data mining engine into SQL Server 2000, which will make panning for gold as easy as writing a SQL Server query.

In the past, data mining tools used different data formats from those available in relational or OLAP (multidimensional) database systems. The data mining extensions in SQL Server 2000 will provide a common format for applications such as statistical analysis, pattern recognition, data prediction and segmentation methods, and visualization products.

The data mining engine in SQL Server 2000 is a powerful platform. Though it ships with two algorithms, it is extensible and supports data mining algorithms that you might build. For instance, if someone came along with a cool algorithm to predict which stocks would do well, they could plug in and leverage this algorithm into the OLE DB for Data Mining API in SQL Server. In addition, the algorithms already integrated into SQL Server 2000 will make it easy for developers using the Visual Basic® development system and SQL Server 2000 to use data mining for analysis, prediction, and reporting purposes, without having to consult a specialist in data mining.

The two algorithms shipped with SQL Server 2000 are a scalable decision tree algorithm and a scalable clustering algorithm. A decision tree algorithm is meant to solve prediction problems. For instance, you might want to predict whether a high school student is going to go to college. If you have a database that contains information about people who did and did not go to college, the decision tree algorithm can use this data to learn rules to make,

Predictions about new input. The rules can also tell you the percentage of probability of the prediction occurring.

Another nice thing about decision trees is that they're interpretable. Someone using the system will be able to tell what rules where used to determine the prediction. Other predictive modeling methods, such as neural networks, are a bit more like a crystal ball, you just feed in the data and the prediction magically appears.

Fast handling of large data sets

Even though decision tree algorithms have been around for a while, they haven't worked well if the training data set is too large. Members of Microsoft Research and members of the SQL Server team at Microsoft came up with some clever techniques to pull the data out of SQL Server and quickly build decision trees from large sets of data.

The clustering algorithms identify maximum similarities within a group, as well as maximum differences between groups. Customers may be grouped or segmented into those most likely to buy a certain product at certain times and under certain conditions. The resulting grouped clients are called clusters. Online stores who cluster their clients will recommend products to their customers based on past purchases. They may recommend you buy products that other people in your cluster have bought, guessing that your predilection for buying hot pink fuzzy slippers means you'll also want the lime-green leg warmers-since the clustering algorithm showed that 9 out of 10 other hot pink fuzzy slipper buyers did.

Statisticians have known about clustering algorithms for decades, however, most of the popular algorithms that are easy to implement will run quickly over small sets of data, but break down when applied to large sets. The main problem is in the design. The algorithms run over and over until the groupings are found, and they may require many scans of the database at each iteration. If a business has a large database or one that is spread out over different servers, pulling the data together for even one customer is non-trivial. It could take days to obtain information about your market clusters.

The scalable clustering algorithm in SQL Server 2000 clusters the database with one scan of the data. This helps address the computational difficulties of collecting data spread throughout an organization on different servers, since the data needs to be read only once.

Right now there's no standard way to cluster customer information that is spread across databases. If a database supports OLE DB for Data Mining, you only need to specify once how to pull everything together. After that, the data mining specification does it for you.

The algorithms also solve problems with high dimensional, sparse data. High dimensional data contains millions of data points in thousands of dimensions. Sparse data means that each entity has only a few of the characteristics that are being measured. Clustering documents is one application of this algorithm.

Chapter 2

Structured Query Language (SQL)

Basics of SQL

Steps to Create a Database

Query with the Database

Link between Two Databases

S.Q.L

SQL (Structured Query Language) is a database sublanguage for querying and modifying relational databases It was developed by IBM Research in the mid 70s and standardized by ANSI in 1986.

The Relational Model defines two root languages for accessing a relational database -- Relational Algebra and Relational Calculus. Relational Algebra is a low-level, operator-oriented language. Creating a query in Relational Algebra involves combining relational operators using algebraic notation. Relational Calculus is a high-level, declarative language. Creating a query in Relational Calculus involves describing what results are desired.

SQL is a version of Relational Calculus. The basic structure in SQL is the statement.

Semicolons separate multiple SQL statements.

There are 3 basic categories of SQL Statements:

SQL-Data Statements -- query and modify tables and columns

o SELECT Statement -- query tables and views in the database,

o INSERT Statement -- add rows to tables,

o UPDATE Statement -- modify columns in table rows,

o DELETE Statement -- remove rows from tables,

o CREATE TABLE Statement -- create tables,

o CREATE VIEW Statement -- create views,

STEPS TO CREATE A DATA BASE :

We use **CREATE TABLE** statement to create a data base. The use of **CREATE** statement is shown below:

CREATE TABLE <table name>

(<column 1 name > <data type>(<size>),
<column 2 name > <data type>(<size>),
<column 3 name > <data type>(<size>));

For example:

CREATE TABLE employee

(emp no int, name char (20), salary int)

The above statement will create a data base or a table named “employee” and there will be thee column naming “ emp-no,name ,salary “, of this data base.

We can even use some constraint to this **CREATE TABLE** Statement like-

1) If we want to give a default value of a particular column of all row we can write

CREATE TABLE employee
(emp no int DEFAULT =1 ,name char (20) DEFAULT = “employee”;

2) If we want to make sure that a particular column of each row cant be left blank or null ,we can write

CREATE TABLE employee
(emp no int NOT NULL, name char (20) NOT NULL);

3) If we want to make sure that the data of a particular column of each row should lie within a particular range

CREATE TABLE employee
(emp no int check (emp no >=1 and emp no <=20);

OR

CREATE TABLE employee
(emp no int check (emp no between 1 and 20);

STATEMENT USED TO MANAGE THE DATA OF A DATA BASE:

To insert the value of a particular row of an already created data base we use:

INSERT INTO <table name > (<column 1 name>, <column 2 name>, ……)
VALUES (<column 1 name>, <column 2 name>, ……)

For example .if we want to insert a data to the table employee, we can write-

INSERT INTO employee (emp no, name, salary)
VALUES (5, 'JOHN', 5000);

2) If we want to update the data of a particular column of a particular row we use UPDATE statement :

The command is:

UPDATE <table name > SET <set.list>
Where <condition>

For example:

UPDATE employee SET SALARY =6000 where emp no =4;

3) If we want to delete a particular row we use :

DELETE FROM (<table name> where < condition>

For example:

DELETE FROM employee WHERE salary = 5000;

The above command will delete all the rows consisting of a “salary“ column of 5000.

The above command will delete all the rows consisting of a “salary“ column of 5000.

QUERY WITH THE DATA BASE

To make query with a database we generally use SELECT FROM statement .This statement has a huge number of separate applications in order to fulfill the random query of the user .This works with an already created data base . The basic use of statement is as follows:

SELECT <column name> FROM < table name>;

Some particular use of SELECT statement is shown below:

1. To see all the data of table employee we give the command

SELECT * FROM employee ;

2. To see all the data of a particular column we use:

SELECT ALL <column name> FROM < table name>;

3. We can have some conditional queries from the database.This condition is satisfied using WHERE statement.

a) If we want to make a query about the data of a particular row having a salary more than 5000,the command is

SELECT * FROM employee where salary >5000;

b) If we want to see a row having a particular name

SELECT * FROM employee where name (‘JOHN SMITH’);

c) To see rows having the value of a particular column using a particular range

SELECT * FROM employee where salary between 2000 and 5000;

d) To see the distinct name of employee

SELECT DISTINCT name FROM employee

4) Some query to match a particular pattern can also be done using matching commands and using LIKE statement

Pattern matches :

1. ampersand(%) matches sub string
2. (-) matches any character
3. Pattern are case sensitive
4. “ % idge matches “ridge”,bridge”.

5. “----“ matches exactly 4 character

6. “----%”matches any string having at least 3 character

a) To see the rows having particular name

SELECT emp no ,name age ,salary FROM employee where name LIKE “JX”;

The above command will match name like JOHN ,JACK

SELECT emp no ,name age ,salary FROM employee where name LIKE “---d”;

The above command will match name like “Brad ,Road”

5) We can see the data or rows in ascending or descending order. The command consist of ORDER BY statement .

SELECT * FROM employee ORDER BY salary ASC ;

The above statement will show all rows of employee data base according to salary in ascending order

LINKING OF TWO OR MORE DATABASES

We can link two or more data bases using appropriate SQL command .The essential condition for linking is that the two database must have a common column through which we can link them.

In order that ,we have to define the common member of the first data base as "**FOREIGN KEY"** and that of the second data base as "**PRIMARY KEY"**

Example:

TABLE – EMPOYEE

Employee no.	Name	Salary	Dept
7369	Sandip	2000	25
7432	Sabir	5000	29
8415	Sayan	6000	21

TABLE –DATA

Dept no	Job	Location
29	Executive	kolkata
25	Manager	New delhi
21	Secretary	Bombay

Hence the common column is Dept of table employee and Dept no. of table data.

To do this we have to define "Dept "column of table employee as foreign key and "Dept no" column of table data as primary key .

CREATE TABLE employee (employee no int ,name char (20), sal int, Dept int FOREIGN KEY);

CREATE TABLE employee (Dept no ,PRIMARY KEY ,Job char (20),location char(20));

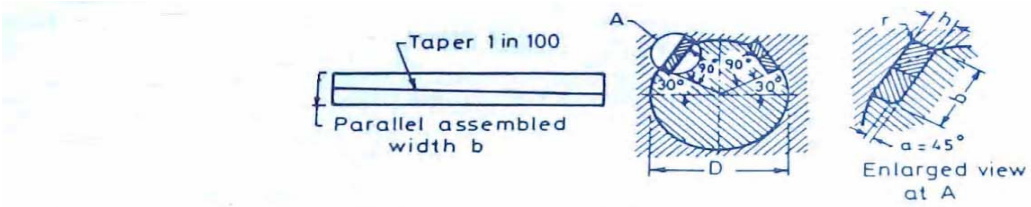
For a particular query,

SELECT employee employee no, employee name, data.job data location FROM employee ,data where employee dept= data dept no. and where employee salary =5000;

The above command will give result as

Employee	Name	Job	Location
7432	Sabir	Executive	Kolkata

We have created a number of database related to machine design elements, so that we can retrieve the data from that particular database as per our requirements.



Shaft Dia. D	*Keyway*			*Key Chamfer a*	*Shaft Dia. D*	*Keyway*			*Key Chamfer a*
	Height h	*Width b*	*Radius r*			*Height h*	*Width b*	*Radius r*	
100	10	30	2	3	460	46	138	4	5
110	11	30	2	3	480	48	144	5	6
120	12	36	2	3	500	50	150	5	6
130	13	39	2	3	520	52	156	5	6
140	14	42	2	3	540	54	162	5	6
150	15	45	2	3	560	56	168	5	6
160	16	48	2	3	580	58	174	5	6
170	17	51	2	3	600	60	180	6	7
180	18	54	2	3	620	62	186	6	7
190	19	57	2	3	640	64	192	6	7
200	20	60	2	3	660	66	198	6	7
210	21	63	2	3	680	68	204	6	7
220	22	66	2	3	700	70	210	6	7
230	23	69	3	4	720	72	216	6	7
240	24	72	3	4	740	74	222	6	7
250	25	75	3	4	760	76	228	6	7
260	26	78	3	4	780	78	234	6	7
270	27	81	3	4	800	80	240	6	7
280	28	84	3	4	820	82	246	6	7
290	29	87	3	4	840	84	252	6	7
300	30	90	3	4	860	86	258	6	7
320	32	96	3	4	880	88	264	8	9
340	34	102	3	4	900	90	270	8	9
360	36	108	3	4	920	92	276	8	9
380	38	114	4	5	940	94	282	8	9
400	40	120	4	5	960	96	288	8	9
420	42	126	4	5	980	98	294	8	9
440	44	142	4	5	1000	100	300	8	9

The above table can be used to find out the required data for keyways in shaft. Only by giving input the shaft diameter we can retrieve required data for the keyway.

To create the database-

CREATE TABLE keyway (shaft_dia int, height int, width int, radius int, chamfer int);

To insert the values into the database-

INSERT INTO keyway (shaft_dia, height, width, radius, chamfer) VALUES (100,10,30,2,3);

INSERT INTO keyway (shaft_dia, height, width, radius, chamfer) VALUES (110,11,30,2,3);

And so on…….

To retrieve the data for any particular shaft diameter-

SELECT FROM keyway WHERE shaft_dia=140;

The above statement will give the output-

Shaft_dia	height	width	radius	chamfer
140	14	42	2	3

Again,

For more example, we have created another database for unidirectional clutches so as to retrieve all the required data, only by inserting the torque to be transmitted.

To create that database-

CREATE TABLE clutches (torque float, inside_dia int, outside_dia int, roller_dia float, length float, length int, number int, roller_pin_dia int, a float);

To insert data into “clutch” database-

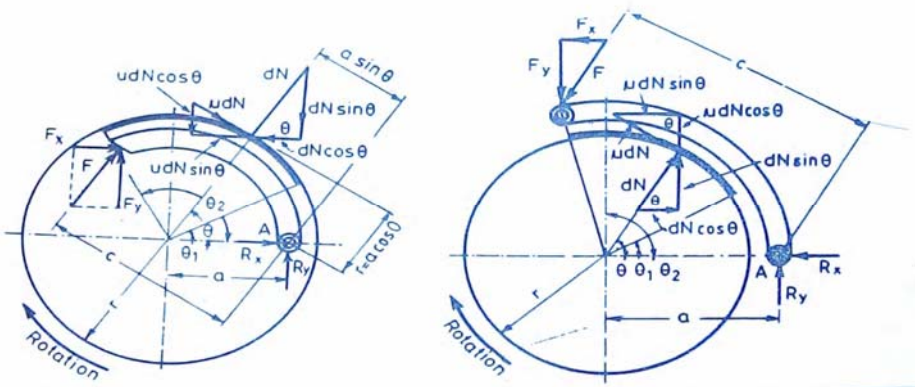
INSERT INTO clutches (torque, inside_dia, outside_dia, roller_dia, length, number, roller_pin_dia, a) VALUES (0.063, 40, 55, 5, 8, 3, 3, 14.85);

Maximum Torque. T, $Kg_f mm$	Inside diameter D-mm.	Outside diameter D_0, mm.	Rollers			Diameter of Roller pin d_1, mm.	a, mm.
			Dia. d-mm.	length l, mm.	Number n		
0.063	40	55	5	8	3	3	14.85
0.0125	40	55	6.5	8	3	3	13.34
0.25	50	65	8	12	3	4	16.80
0.4	50	65	8	12	5	4	18.77
0.63	56	70	9	14	5	4	22.70
1.0	70	85	11	15	5	5	23.70
1.6	70	85	12	18	6	6	30.60
2.5	90	110	14	20	6	6	33.60
4.0	100	120	16	24	6	8	33.60
6.3	120	140	16	24	8	8	43.50
10.0	140	165	19	28	8	8	50.40
16.0	160	190	22	34	8	8	57.30

To find all the required data only by giving a particular "torque", we can use—

SELECT * FROM clutches WHERE torque=1.6;

The output will be-



Torque	inside_dia	outside_dia	roller_dia	length	number	roller_pin_dia	a
1.6	70	85	12	18	6	6	30.60

Even within a SQL database, we can insert diagram of any particular machine parts, just by giving the file name which contain that diagram.

Chapter 3

ASP.NET

Global.asa

Nimstyle.css

Nitrdesign.asp

Nitrelements.asp

Mainpage.html

ASP.NET PROGRAMMING

Asp.net is a .net programming which can be run through Internet Information System (IIS) in WINDOWS 2003 Enterprise Edition and it is the programming through which we can interact with the existing database. Actually asp.net is used to produce the user interface of the Data Mining system. All the programming codes here is written in NOTEPAD and is saved with .asp/.aspx extension. Our aim was to produce a number of asp.net pages, combination of which enables the user to get the connectivity to the database and to retrieve the information as per his requirements. The pages along their description are given below-

global.asa:

This is the asp.net programming code in which is responsible for establishing connection to the Database. It consist of the server name, its own user id and password along with the Database name. The code is following:

```
<SCRIPT LANGUAGE="VBScript" RUNAT="Server">

Sub Session_OnStart    '  Runs the first time a user runs any page in your application
msgbox "test"

        'Create an ADO Connection

        Set cn = Server.CreateObject("ADODB.Connection")

        ' Specify the OLE DB provider.

        cn.Provider = "sqloledb"

        ' Specify connection string on Open method.

        ' To run the sample, edit this line to reflect your server's name.

        'ProvStr = "Server=JHAKAS;Database=Design;UID=sa;PWD=;"

         ProvStr = "Server=JHAKAS;Database=Design;"

        cn.Open ProvStr msgbox "test"

        Set Session("cnn") = cn

END Sub
```

```
Sub Session_OnEnd    '   Runs when a user's session times out or quits your application

        'Close the data connection

        cn.Close

END Sub

</SCRIPT>
```

Nimstyle.css:

It is responsible for setting the background and font colour to the particular asp or html page. The programming code is following:

```
BODY
{
   BACKGROUND-COLOR: PINK;
   COLOR: black;
   FONT-FAMILY: verdana, arial, times new roman;
   FONT-SIZE: 10pt;
   FONT-WEIGHT: bold
}

H2
{
   COLOR: purple
}
H3
{
   COLOR: black
}
TABLE
{
   COLOR: black;
   FONT-FAMILY: arial;
   FONT-SIZE: 10pt;
   FONT-WEIGHT: normal
}
.Header
{
   BACKGROUND-COLOR: teal;
   COLOR: yellow;
   FONT-FAMILY: verdana, arial;
   FONT-WEIGHT: bold
}
.formtable
{
   BACKGROUND-COLOR: LIGHTBLUE}
```

```
TD
{
   BACKGROUND-COLOR: LIGHTBLUE
}
H4
{
   COLOR: black;
   FONT-FAMILY: arial;
   FONT-SIZE: 20pt;
   FONT-WEIGHT: bold

}
H6
{
   COLOR: red
}
```

nitrdesign.htm:

This is the html page, which ask the user to enter the element to retrieve the design elements. The input is passed on to “nitrelements.asp” file. The code is following:

```
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Developer Studio">
<META HTTP-EQUBODY
{
   BACKGROUND-COLOR: PINK;
   COLOR: black;
   FONT-FAMILY: verdana, arial, times new roman;
   FONT-SIZE: 10pt;
   FONT-WEIGHT: bold
}
H2
{
   COLOR: purple
}
H3
{
   COLOR: black
}
TABLE
{
   COLOR: black;
   FONT-FAMILY: arial;
   FONT-SIZE: 10pt;
   FONT-WEIGHT: normal
}
.Header
```

```
{
   BACKGROUND-COLOR: teal;
   COLOR: yellow;
   FONT-FAMILY: verdana, arial;
   FONT-WEIGHT: bold
}
.formtable
{
   BACKGROUND-COLOR: LIGHTBLUE
}
TD
{
   BACKGROUND-COLOR: LIGHTBLUE
}
H4
{
   COLOR: black;
   FONT-FAMILY: arial;
   FONT-SIZE: 20pt;
   FONT-WEIGHT: bold

}
H6
{
   COLOR: red
}
IV="Content-Type" content="text/html; charset=iso-8859-1">
<TITLE>NITR ONLINE Design Database</TITLE>
<LINK REL="stylesheet" TYPE="text/css" HREF="nimstyle.css">
</HEAD>
<BODY>

<CENTER>
<H2>NITR ONLINE Database</H2>
<HR>
<H3></H3>
</CENTER>

<center>
<form method="post" action = "nitrelements.asp" >
TYPE elements <input type="text" name="name" size="20">

<input type="Submit" value="Submit">
</form>


</BODY>
</HTML>
```

Nitrelements.asp:

This is the asp.net programming code in which the input from the previous file is sent to the Database inform of a query and the values are retrieved. The programming code is following:

```
<%@ LANGUAGE="VBSCRIPT" %>

<HEAD>

<META NAME="GENERATOR" Content="Microsoft Developer Studio">

<META HTTP-EQUIV="Content-Type" content="text/html; charset=iso-8859-1">

<TITLE>Northwind Inventory Management</TITLE>

<LINK REL="stylesheet" TYPE="text/css" HREF="nimstyle.css">

</HEAD>

<HTML>
<head><title>Responding to a form</title>

<SCRIPT LANGUAGE="VBScript" RUNAT="Server">

</SCRIPT>

</head>

<body>

<%
Dim StartTime, EndTime, TimeIt, tek1

StartTime = Timer

tek1=Request.Form("name")

'response.write(tek1)
response.write("<br />")

%>
<%
'Get the display operation from the querystring

'    The operation determines whether the product information display

'    will contain Update or Remove options.

op = Request.QueryString("op")
```

```
'Reference the Session connection variable
cn = Session("cnn")

'Create a recordset
Set CategoryInfo = Server.CreateObject("ADODB.Recordset")

'Create a query string
response.write("<br />")


dim  query_new1
query_new1=" "

query_new1= "SELECT elements, values FROM database WHERE elements = '"&tek1
query_new1= query_new1 & "'"
'response.write query_new1
querystr=query_new1
msgbox querystr

'Open the Recordset with the query string on the connection cn.
CategoryInfo.Open querystr, cn
%>

<!-- Output Page Header -->
<CENTER>
<H2>elements</H2>
<HR>
<!-- Output the table of categories and their descriptions -->
<TABLE  CELLPADDING=3 BORDER=0 COLSPAN=8>
        <TR>
                <% 'Output the table headers
                For i = 0 To 1
                        Response.Write "<TD CLASS=header>" &
CategoryInfo.Fields(i).Name & "</TD>"
                NEXT %>
        </TR>
        <% 'Loop through the recordset and output the table values
        Do While Not CategoryInfo.EOF
                Response.Write "<TR>"
                For i = 0 to 1
                        Response.Write "<TD>"

                                Response.Write CategoryInfo.Fields(i).Value

                        Response.Write "</TD>"
                NEXT
                Response.Write "</TR>"
                CategoryInfo.MoveNext
        LOOP %>
</TABLE>
```

```
<%      'Close the recordset
        CategoryInfo.Close %>

</CENTER>
<BR><BR><HR>
Return to the <A HREF="intro.htm">Main Menu</A>

</BODY></HTML>
```

main page:

This is the main page coded to interact with the user directly. The input inserted by user is sent to “nitrdesign.asp” and the output information after processing is shown in this page. The programming code is following:

```
<html>
<head><title>desisn</title></HEAD
<body bgcolor=black>
<center><font color=black size=12><b><u><i>N.I.T   ROURKELA DESIGN DATABASE
</b></i></font></u></center><br>

<hr size=5 width="100%" valign=left color =red>
<br>
<center>
<a href="images.jpg"><img src="images.jpg" height=80 width=100
align=center></a></center>

<br>
<br>
<center>
<a href="engin.jpg"><img src="engin.jpg" height=80 width=100 align=center></a>
<a href="gear.jpg"><img src="gear.jpg" height=80 width=100 align=center></a>
<a href="turbine.jpg"><img src="turbine.jpg" height=80 width=100 align=center></a>
<a href="turbine2.jpg"><img src="turbine2.jpg" height=80 width=100 align=center></a>
<a href="lathe2.jpg"><img src="lathe2.jpg" height=80 width=100
align=center></a></center>

<br>
<br>
<br>

<center>
<font color=black><h3>Enter your keywords here:-

<input type="text" name="x" size=15><br>
</center>
<br>
<center>
<font color=black><h3>Select your required table:-
<select name="name">
```

```
<option>getValue(query_new1=select elements, values from database where name(i)=”” 
tek1
<option>getValue(query_new1=select elements, values from database where name(i)=”” 
tek1
<option>getValue(query_new1=select elements, values from database where name(i)=”” 
tek1
</select>
<center>
<br>
<br>
<br>


<table bgcolor=silver border cellspacing=1><center><caption><font color=voilet> Enter any 
value and click on submit:</caption></center></font>


<th><font color=black>Request.QueryString(“nitrelements.asp”) <th><font color=black> 
Request.QueryString(“nitrelements.asp”)  <th><font color=black> 
Request.QueryString(“nitrelements.asp”)  </font> <th><font color=black> 
Request.QueryString(“nitrelements.asp”) <th><font color=black> 
Request.QueryString(“nitrelements.asp”) <th>

<tr colspacilng=2><td ><input type="text" name="x" size=15> 
<td><input type="text" name="x" size=15><td ><input type="text" name="x" size=15><td 
><input type="text" name="x" size=15>
<td><input type="text" name="x" size=15>
</table>
<br>

<input type="button" value="SUBMIT"  onClick="doform(this.form)">
<input type="reset" name="reset_button" value="RESET">
</body>
</html>
```

Chapter 4

CONCLUSION

Results and Output

Scope for further enhancement

Conclusion

References

RESULTS and OUTPUT

The web page of the output is shown in the following two pages. In that webpage, the user is first asked for the keyword related to the required table name. If user give the keyword and click on “enter”, the page will provide a list of the related tables on the next row. Then on selecting the required table name, a related table with a blank row will be provided. Now if the user inserts the value of any row and then click on submit, the software will provide the data of the rest columns.

An example is shown on the next page,

Here if we type “key”, then software will provide four options, like: 1) keyhole,
2) taper key,
3) gib-head key,
4) woodruff key,

On selecting “keyhole”, the software will provide the table shown on next to next page, now insert the value in any column and click on “submit”.

The output table is shown on the same page.

N.I.T ROURKELA DESIGN DATABASE

Enter your keywords here:- key

Select your required table:-

N.I.T ROURKELA DESIGN DATABASE

Enter your keywords here:- key

Select your required table:- keyw ay

Enter any value and click on submit:				
Shaft dia	**height**	**width**	**radius**	**key chamfer**

SUBMIT

Scope for the Future

Data Mining is simply a vast concept. Through this project we have only proved that Data Mining process with the help of Artificial intelligence can be used for the design in mechanical industries. There are lots more things that can be applied to make the Data Mining process more efficient:

1) The Database can be made a huge one, consisting of every design elements to make it a world design data cell.
2) The user interface can be made more efficient, more intelligent and more user friendly.
3) Some applications can also be developed in other Database like: Oracle, MS-ACCESS etc, using JAVA.
4) The Data Mining process can be improved to contain every design data for a mechanical industry.

CONCLUSION

It is found that the interfaced developed works successfully. Thus it is proved that the data retrieval system using data mining and A.I techniques work in case of a data base management system. The same interface can be used in mechanical engineering industry.

A few applications are given below:

i. **Production Process**: Consider the example of production of a lathe. We know that most industries depend on paper work to determine and maintain records of bill of materials which is the integral part of any production process. If a database of the same is developed, it becomes easy to retrieve requisite data ,like in this case ,material required for the bed, who are the suppliers ,its cost, its current availability etc are all just as near as a click of a mouse. Unlike paper work there is no threat to loss of information, since the computer field is well equipped with adequate backing up facilities. Also the delay in paper work decreases and improves the efficiency of the industry

ii. **Maintenance:** Minimized paper work results in a efficient maintenance facility. The lesser the delay the better will be the working of the different machines. Considering an example in the maintenance field, a refinery which has on an average around 5000 rotary equipment. If the equipment configuration and previous failure list is available with the chief engineer, he can see to it that all equipments are in good working condition on the report of a fault it becomes easy for him to analyze the failure causes and takes appropriate steps to correct them.

iii. **Employee list management:** This is a managerial concern and using this interface it becomes easy to maintain employee record, their field of specialization and through it the benefits they should be getting.

iv. **Forecasting:** Using artificial intelligence it is easy to project accuracy set of variables to a future value. So by feeding the present values into a database, it becomes easy to set targets, allocate time inputs & manage monetary concerns.

REFERENCES

Books:-

1) Information Technology,
by Sumit Arora,

2) HAND BOOK of Properties Engineering Materials and Design Data for Machine elements
by Abdullah Shariff,

Websites:-

1) www.mssql.com,
2) www.the-data-mine.com,
3) www.codeproject.com,
4) www.aspnetpro.com,
5) www.wikipedia.org,
6) www.the-data-mine.com,